REMARKS

I. Introduction

In response to the Office Action dated March 30, 2006, the claims have not been amended. Claims 1-10, 12-15, 17-25, 27-40, 42-45, 47-55, and 57-59 remain in the application. Re-examination and re-consideration of the application is requested.

II. Non-Art Rejections

In paragraph (4) of the Office Action, claims 12, 13, 27, 28, 42, 43, 57, and 38 were rejected under 35 U.S.C. §112, first paragraph, as failing to comply with the enablement requirement. In paragraph (5) of the Office Action, claims 12, 13, 27, 28, 42, 43, 57, and 58 were rejected under 35 U.S.C. §112, second paragraph, as being incomplete for omitting essential structural cooperative relationships of elements, such omission amounting to a gap between the necessary structural connections.

Applicants respectfully traverse the above rejections.

With respect to claims 12, 13, 27, 28, 42, 43, 57, and 58, the claims were rejected stating that while the specification suggests the use of multiplexors with the hardware state machine, nowhere is it described in the manner in which the multiplexors are actually used or configured. The rejections further state that it is unclear how the multiplexors relate to the remainder of the invention. Applicants refer the Patent Office to paragraphs [0070]-[0074] that provide as follows:

> **[0070]** The hardware state machine 718 may contain the same logic as used in the prior art and may not be modified. In addition to the state machine 718, the implementation consists of a permutation that employs a series of configurable multiplexors at the beginning 716 and end 720 of the fixed hardware state machine 718. Custom logic (i.e., the logic within hardware configuration control and IO module 714) interconnects the multiplexors (within permutations 716 and 720) to the system bus 612 of CAM 512. Accordingly, the hardware configuration control and IO module 714 that connects to the system bus 612 controls access to the permutation 716 and 720 and state machine 718 logic.
> **[0071]** The custom logic within the control and IO module 714 implements a key exchange protocol by accepting (or rejecting) a series of pre-authorized keys (e.g., sequentially wrapped keys with $n=10^6$ times or other large value) or other secure protocol. The key defines a configuration for the permutations 716 and 720. Valid keys are only known to the headend (e.g., uplink center 104) by using any public key algorithm such as Rabin or RSA (Rivest-Shamir-Adelman). Based on a public key algorithm, the keys cannot be recreated or generated by unknown parties. The keys are delivered to the smart card either over the broadcast stream, Internet, or other

appropriate distribution channel. The keys can be delivered to the smart card (i.e., CAM 512) population asynchronously (e.g., over a period of several hours, days, or months). The keys may be delivered using uniquely encrypted, group encrypted packets. These packets are unintelligible to members (i.e., CAMs 512) for which they were not encrypted. In other words, the packets are only intelligible to those members/control and IO modules 714 having the appropriate private key.

[0072] The hardware configuration control and IO module 714 verifies/authenticates the key. Such a verification/authentication may ensure that the key is from a known source (e.g., a known uplink center 104, program source 200A-200C,etc.), that the key is not a duplicate of an already received key, or that the key fails to comply with an additional security measure. As part of the authentication process, the control and IO module 714 decrypts the keys. The decrypted key is then verified/authenticated by the custom logic within module 714. If the key is valid, the key is retained by the control and IO module 714 (e.g., by storing the keys in protected registers with no physical or logical output mechanism outside the custom logic within the module 714). If the key is invalid, the key is rejected and may not be stored by the control and IO module 714.

[0073] As described above, the key defines a configuration for the permutations 716 and 720. Accordingly, when appropriate, the key is used to dynamically (i.e., on-the-fly) reconfigure the permutations 716 and 720. The timing of the reconfiguration may occur immediately upon receipt of the key. Alternatively, the key may be stored by control and IO module 714 and only used to reconfigure the permutations 716 and 720 (e.g., switch the configuration to that represented by the stored key) upon receipt of an over the air command. In such a circumstance, the control and IO module 714 may store a currently active key (that defines a permutation 716 and 720 currently being used) and a future key. Accordingly, the keys may be delivered asynchronously over a very long period of time to multiple CAMs 512 where they are validated and stored asynchronously. Thereafter (e.g., once a period of time has passed to ensure that appropriate/enough CAMs 512 have the new key), an over the air command to activate a reconfiguration operation for a key may be delivered synchronously to all CAMs 512. Thus, the actual reconfiguration operation may occur simultaneously within all CAMs 512, while the key delivery and validation mechanism is asynchronous over a period of time.

[0074] To recongifure the permutations, 716 and 720, the control and IO module 714 communicates bi-directly 722 with the pre-permutations 716 and post-permutations 720 to dynamically configure the series of multiplexors in each respective permutation 716 and 720. Once configured, the pre-permutations 716 place the digital services information received across communication link 724 from control and IO module 714 into the appropriate form for use by the hardware state machine 718. Hardware state machine 718 may modify the digital services information based on custom logic within the state machine 718. Thereafter, the post-permutations 720 may modify the outgoing digital services information to limit use and viewing of the information from unauthorized attackers.

The *The American Heritage® Dictionary of the English Language, Fourth Edition Copyright © 2000* (see www.dictionary.com) defines permutation as "A rearrangement of the elements of a set."

Further, a multiplexor is merely a hardware device having one or more inputs and one or more outputs where the mulitplexor determines the signal that is output. As set forth in the specification above, the control and IO module 714 communicates with the permutations to dynamically configure the series of multiplexors in each permutation. The specification further provides, that once configured, the pre-permutation places the digital services received from the control and IO module 714 into the appropriate form for the hardware state machine 718. Thereafter, the post-permutation modifies the outgoing digital services information to limit the use and viewing of the information from unauthorized attackers (see paragraph [0074]). Thus, the specification clearly describes the manner in which the multiplexors are configured and used within/by permutations. In addition, as set forth in the specification, the keys that define the configuration are used to dynamically reconfigure the permutations (and the multiplexors within the permutations). Further, since the multiplexors are employed by the permutations and are configurable, the specification clearly supports and enables the use of such multiplexors to one with knowledge in the field of art.

In response to the above arguments, the Office Action recites the Microsoft Computer Dictionary and states that the known functionalities of a multiplexor do not result in the permuting of an input. Namely, the Office Action asserts that a multiplexor is used to either attach many communication lines to a smaller number of communications ports or to attach a large number of communications ports to a smaller number of communications lines. Applicants note that a copy of the actual definition was not provided to Applicants not recited as a reference. Nonetheless, Applicants submit that based on this definition, a multiplexor merely receives one or more inputs and determines the appropriate output. It is not required to reduce the input to a smaller number of communications ports or to increase it to a larger number of communications ports. It is merely used to take input and produce an output – smaller, larger, or the same number.

A permutation is defined as the rearrangement of inputs. In this regard, a permutation can be performed by a multiplexor simply by rearranging the input lines to different output lines. In fact, even using the definition provided by the Examiner, the input lines would have to be modified or permuted to produce the output lines. Thus, the permutations described in the specification do in fact enable the use of multiplexors as described therein. Further, the control and IO module 714 is configured to dynamically configure the multiplexors into the desired configuration.

The Office Action asserts that the specification paragraph 70 simply states that the invention connects multiplexers using custom logic. Applicants respectfully disagree and traverse such an assertion. Paragraph 70 is recited above and provides that a permutation employs a series of

configurable multiplexors. Further, contrary to that asserted in the Office Action, paragraph 70 states that the custom logic interconnects the configurable multiplexors to the system bus of the CAM. Again, the custom logic utilizes these configurable multiplexors at the beginning and end of the hardware state machine. Further, the above paragraphs specify that a key is used to define the configuration for the multiplexors. The above paragraphs 70-74 provide and describe explicitly how the multiplexors are used in the invention and in fact provide enabling support under 35 U.S.C. 112.

The Office Action asserts that there would be many designs for permutors that are well known in the art. Such a statement in itself is an admission that the multiplexors are enabled. One cannot assert that it is well known to use multiplexors to produce a desired configuration yet also argue that such a use is not enabling. The Action then asserts that it would require undue experimentation to implement a particular arrangement of multiplexors. Applicants note that above cited paragraphs specify the use of keys that are used to configure the multiplexors. Such a definition in combination with the known art use of multiplexors clearly enables the invention.

In view of the above, Applicants respectfully request that the rejections under 35 U.S.C. 112 be withdrawn.

III. Prior Art Rejections

In paragraph (6) of the Office Action, claims 15, 17-19, 21-24, 30-34, 36-39, 45, 47-49, and 52-54 were rejected under 35 U.S.C. §102(b) as being anticipated by Campinos et al. (Campinos), U.S. Patent No. 6,035,038. In paragraph (7) of the Office Action, claims 1-10, 12, 15, 17-25, 27, 30-40, 42, 45, 47-55, and 57 were rejected under 35 U.S.C. §103(a) as being unpatentable over Campinos, in view of Wasilewski et al. (Wasilewski), U.S. Patent No. 6,157,719.

In paragraph (8) of the Office Action, claims 13, 28, 43, and 58 were rejected under 35 U.S.C. §103(a) as being unpatentable over Campinos, in view of Wasilewski, and further in view of Killian, U.S. Patent No. 5,222,141. However, in paragraphs (9)-(10) of the Office Action, claims 14, 29, 44, and 59 were indicated as being allowable if rewritten in independent form to include the base claim and any intervening claims.

Applicants acknowledge the indication of allowable claims, but respectfully traverse these rejections.

Specifically, claims 1, 15, 30, and 45 were rejected as follows:

> As to claims 15, 18, 30, 31, 33, 45, and 48, Campinos discloses a conditional access system wherein a security component implemented on a smartcard is used to decipher asynchronously transmitted entitlement messages for controlling access in an access control unit. The access control

unit is not directly accessible to the system bus, and the smartcard has no local bus (see figure 5 and column 5, lines 21-67).

As to claims 1, 2, 15, 30, 31, and 45, Wasilewski discloses an access system for set-top boxes wherein configuration information may be transmitted to the set-top box as a one-time event (i.e. asynchronously). Since the set-top box's function is to determine whether an encrypted instance should be decrypted, it constitutes a security component that controls access to digital services. The received configuration information (the EMM) comprises decryption keys (control words) to be implemented (see column 6, line 24 to column 7, line 24) in a hardware state machine (the DHCT) such as an ASIC (see column 15, lines 32-36 and figures 2B and 3). A control suite (the control center) sends transmissions via satellite, which inherently employs an uplink center for sending transmissions to the satellite. The stream is incorporated at a media server for distribution (see column 15, lines 7-24). The system comprises a smart card (see column 21, line 13).

The components of the DHCTSE, which contains the hardware state machine, are only accessible to the system bus or I/O via the DHCT interface. In Wasilewski's implementation, components of the DHCTSE communicate with one another via a local bus (see figure 12 and column 21, lines 15-27).

Campinos discloses a conditional access system wherein access control (the equivalent of the DHCTSE) is preformed on a user smartcard having no local bus and no direct outside access to the access control circuit (see figure 5 and column 5, line 28 to column 6, line 14) and suggests that this card makes it possible to verify that entitlements in the EMM are reserved for the user (see column 3, lines 48-57).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to implement the invention of Wasilewski using the smartcard of Campinos for the DHCTSE, as this card makes it possible to verify that entitlements in the EMM are reserved for the user.

Applicant traverses the above rejections for one or more of the following reasons:

(1) Neither Campinos, Wasilewski, nor Killian teach, disclose or suggest the dynamic reconfiguration of a hardware state machine;

(2) Neither Campinos, Wasilewski, nor Killian teach, disclose or suggest dynamically reconfiguring a hardware state machine based on configuration information that is received asynchronously;

(3) Neither Campinos, Wasilewski, nor Killian teach, disclose or suggest a hardware state machine containing custom logic; and

(4) Neither Campinos, Wasilewski, nor Killian teach, disclose or suggest that a component of the hardware state machine that is not directly accessible to a system input/output module or a system bus of the security component.

Independent claims 1, 15, 30, and 45 are generally directed to the use of a CAM/smart card to prevent unauthorized access to digital services. Specifically, configuration information for accessing digital services is transmitted asynchronously to a subscriber. The CAM/smart card receives the configuration information. The CAM/smart card has many components including a custom logic block and a hardware state machine. The claims provide that the hardware state machine is not directly accessible to a system input/output module or system bus of the

CAM/smart card. Further, the custom logic block is configured to dynamically reconfigure the hardware state machine based on the configuration information that is received. Also, it is noted that the hardware state machine comprises custom logic that is used to control access to the digital services.

Thus, as claimed, since the hardware state machine is not accessible to the system input/output module or system bus of the CAM/smart card, and because the implementation is hardware based, it is protected from being altered by the microprocessor of the CAM/smart card or external means.

In the prior art, hardware was not reconfigurable as claimed. Further, the prior art failed to protect the integrity of the CAM/smart card. In this regard, tin the prior art, software within the CAM/smart card was merely altered through inappropriate manipulation of the microprocessor memory access control unit. However, in the present invention, not only is hardware used, but the hardware is isolated from/not accessible to a system input/output module or system bus (and thereby not directly accessible to the microprocessor).

In rejecting the claims, the Office Action relies on FIG. 5 and col. 5, lines 21-67 of Campinos. FIG. 5 provides:

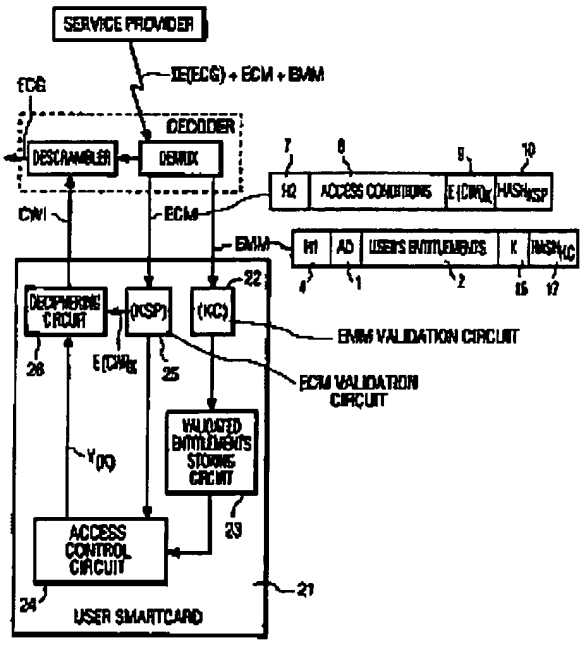

FIG. 5

As can be clearly seen by the above figure, the access control unit 24 is directly accessible to the deciphering circuit 26, the KSP 25, and the validated entitlements storing circuit 23. The way to access such circuits 23, 25, and 26 is via various lines. Such lines are the only possibility to be the system bus in Campinos. An electronic search of Campinos for the term "bus" provides no results. Thus, Campinos either fails to teach or suggest a system bus or the lines connecting the various circuits are being read as equivalent to the system bus. The claims of the present invention explicitly provide that the security component (i.e., the CAM) does in fact have a system bus. Thus, any prior art that fails to show, illustrate, or describe a system bus cannot read on the present invention (nor render it obvious). In view of the above, Applicants submit that since Campinos fails to describe a system bus or even mention the term "bus", Campinos cannot possibly describe the invention. Alternatively, if the various lines connecting the circuits are viewed as the system bus, the access control unit 24 is clearly directly connected to such lines (and thereby fails to read on the claims).

In addition to the above differences, Applicants note that the claims explicitly provide that configuration information is used to dynamically reconfigure a hardware state machine within the security component/smart card. The Office Action completely fails to even acknowledge such a dynamic reconfiguration. Further, Campinos not only fails to describe a hardware state machine but completely fails to describe the ability to dynamically reconfigure such a hardware state machine. In this regard, prior art hardware state machines are often fixed. However, the present invention provides the ability to dynamically reconfigure a hardware state machine using the configuration information that has been transmitted asynchronously. Nowhere in Campinos is there even a remote reference to such configuration information or the ability to use such configuration information to dynamically reconfigure a hardware state machine. In this regard, Campinos' FIG. 5 clearly illustrates a fixed smart card with no ability to dynamically reconfigure any of the hardware states therein.

In addition, the remaining references fail to cure the deficiencies of Campinos.

Moreover, the various elements of Applicants' claimed invention together provide operational advantages over Campinos, Wasilewski, and Killian. In addition, Applicants' invention solves problems not recognized by Campinos, Wasilewski, and Killian.

Thus, Applicants submit that independent claims 1, 15, 30, and 45 are allowable over Campinos, Wasilewski, and Killian. Further, dependent claims 2-10, 12-14, 17-25, 27-29, 31-40, 42-44, 47-55, and 57-59 are submitted to be allowable over Campinos, Wasilewski, and Killian in the same manner, because they are dependent on independent claims 1, 15, 30, and 45, respectively, and

thus contain all the limitations of the independent claims. In addition, dependent claims 2-10, 12-14, 17-25, 27-29, 31-40, 42-44, 47-55, and 57-59 recite additional novel elements not shown by Campinos, Wasilewski, and Killian.

IV. Conclusion

In view of the above, it is submitted that this application is now in good order for allowance and such allowance is respectfully solicited. Should the Examiner believe minor matters still remain that can be resolved in a telephone interview, the Examiner is urged to call Applicants' undersigned attorney.

Respectfully submitted,

By: _____
Name: Georgann S. Grunebach
Reg. No.: 33,179

Date: June 27, 2006

The DIRECTV Group, Inc.
RE/R8/A109
2230 E. Imperial Highway
P. O. Box 956
El Segundo CA 90245

Telephone No. (310) 964-4615